Multiple Testing Procedures

Katherine S. Pollard[1], Sandrine Dudoit[2], Mark J. van der Laan[3]

October 6, 2004

1. Center for Biomolecular Science and Engineering, University of California, Santa Cruz, `http://lowelab.ucsc.edu/katie/`
2. Division of Biostatistics, University of California, Berkeley, `http://www.stat.berkeley.edu/~sandrine/`
3. Department of Statistics and Division of Biostatistics, University of California, Berkeley, `http://www.stat.berkeley.edu/~laan/`

Contents

1 Introduction

1.1 Overview

The Bioconductor R package *multtest* implements widely applicable resampling-based single-step and stepwise multiple testing procedures (MTP) for controlling a broad class of Type I error rates, in testing problems involving general data generating distributions (with arbitrary dependence structures among variables), null hypotheses, and test statistics Dudoit and van der Laan (2004); Dudoit et al. (2004); van der Laan et al. (2004b,a); Pollard and van der Laan (2004). The current version of *multtest* provides MTPs for null hypotheses concerning means, differences in means, and regression parameters in linear and Cox proportional hazards models. Both bootstrap and permutation estimators of the test statistics (t- or F-statistics) null distribution are available. Procedures are provided to control Type I error rates defined as tail probabilities and expected values of arbitrary functions of the numbers of Type I errors, V_n, and rejected hypotheses, R_n. These error rates include: the generalized family-wise error rate, $gFWER(k) = Pr(V_n > k)$, or chance of at least $(k + 1)$ false positives (the special case $k = 0$ corresponds to the usual family-wise error rate, FWER); tail probabilities $TPPFP(q) = Pr(V_n/R_n > q)$ for the proportion of false positives among the rejected hypotheses; the false discovery rate, $FDR = E[V_n/R_n]$. Single-step and step-down common-cut-off (maxT) and common-quantile (minP) procedures, that take into account the joint distribution of the test statistics, are implemented to control the FWER. In addition, augmentation procedures are provided to control the gFWER, TPPFP, and FDR, based on *any* initial FWER-controlling procedure. The results of a multiple testing procedure are summarized using rejection regions for the test statistics, confidence regions for the parameters of interest, and adjusted p-values. The modular design of the *multtest* package allows interested users to readily extend the package's functionality, by inserting additional functions for test statistics and testing procedures. The S4 class/method object-oriented programming approach was adopted to summarize the results of a MTP.

1.2 Motivation

Current statistical inference problems in areas such as genomics, astronomy, and marketing routinely involve the simultaneous test of thousands, or even millions, of null hypotheses. Examples of testing problems in genomics include:

- the identification of differentially expressed genes in microarray experiments, i.e., genes whose expression measures are associated with possibly censored responses or covariates interest;
- tests of association between gene expression measures and Gene Ontology (GO) annotation (`www.geneontology.org`);
- the identification of transcription factor binding sites in ChIP-Chip experiments, where chromatin immunoprecipitation (ChIP) of transcription factor bound DNA is followed by microarray hybridization (Chip) of the IP-enriched DNA Keleş et al. (2004);
- the genetic mapping of complex traits using single nucleotide polymorphisms (SNP).

The above testing problems share the following general characteristics:

- inference for high-dimensional multivariate distributions, with complex and unknown dependence structures among variables;
- broad range of parameters of interest, such as, regression coefficients in model relating patient survival to genome-wide transcript levels or DNA copy numbers, pairwise gene correlations between transcript levels;
- many null hypotheses, in the thousands or even millions;
- complex dependence structures among test statistics, e.g., Gene Ontology directed acyclic graph (DAG).

Motivated by these applications, we have developed resampling-based single-step and step-down multiple testing procedures (MTP) for controlling a broad class of Type I error rates, in testing problems involving general data generating distributions (with arbitrary dependence structures among variables), null hypotheses, and test statistics Dudoit and van der Laan (2004); Dudoit et al. (2004); van der Laan et al. (2004b,a); Pollard and van der Laan (2004). In particular, Dudoit et al. Dudoit et al. (2004) and Pollard & van der Laan Pollard and van der Laan (2004) derive *single-step common-cut-off and common-quantile procedures* for controlling arbitrary parameters of the distribution of the number of Type I errors, such as the generalized family-wise error rate, $gFWER(k)$, or chance of at least $(k+1)$ false positives. van der Laan et al. van der Laan et al. (2004b) focus on control of the family-wise error rate, $FWER = gFWER(0)$, and provide *step-down*

common-cut-off and common-quantile procedures, based on maxima of test statistics (maxT) and minima of unadjusted p-values (minP), respectively. Dudoit & van der Laan Dudoit and van der Laan (2004) and van der Laan et al. van der Laan et al. (2004a) propose a general class of *augmentation multiple testing procedures* (AMTP), obtained by adding suitably chosen null hypotheses to the set of null hypotheses already rejected by an initial MTP. In particular, given *any* FWER-controlling procedure, they show how one can trivially obtain procedures controlling tail probabilities for the number (gFWER) and proportion (TPPFP) of false positives among the rejected hypotheses.
A key feature of our proposed MTPs is the *test statistics null distribution* (rather than data generating null distribution) used to derive rejection regions (i.e., cut-offs) for the test statistics and resulting adjusted p-values Dudoit and van der Laan (2004); Dudoit et al. (2004); van der Laan et al. (2004b,a); Pollard and van der Laan (2004). For general null hypotheses, defined in terms of submodels for the data generating distribution, this null distribution is the asymptotic distribution of the vector of null value shifted and scaled test statistics. Resampling procedures (e.g., based on the non-parametric or model-based bootstrap) are proposed to conveniently obtain consistent estimators of the null distribution and the resulting test statistic cut-offs and adjusted p-values Dudoit et al. (2004); van der Laan et al. (2004b); Pollard and van der Laan (2004).
The Bioconductor R package *multtest* provides software implementations of the above multiple testing procedures.

1.3 Outline

The present vignette provides a summary of our proposed multiple testing procedures (Dudoit and van der Laan (2004); Dudoit et al. (2004); van der Laan et al. (2004b,a); Pollard and van der Laan (2004). Section 2), discusses their software implementation in the Bioconductor R package *multtest* (Section 3). The accompanying vignette (MTPALL) describes their application to the ALL dataset of Chiaretti et al. Chiaretti et al. (2004).
Specifically, given a multivariate dataset (stored as a *matrix*, *data.frame*, or microarray object of class *exprSet*) and user-supplied choices for the test statistics, Type I error rate and its target level, resampling-based estimator of the test statistics null distribution, and procedure for error rate control, the main user-level function `MTP` returns unadjusted and adjusted p-values, cut-off vectors for the test statistics, and estimates and confidence regions for the parameters of interest. Both bootstrap and permutation estimators of

the test statistics null distribution are available and can optionally be output to the user. The variety of models and hypotheses, test statistics, Type I error rates, and MTPs currently implemented are discussed in Section 3.2. The S4 class/method object-oriented programming approach was adopted to represent the results of a MTP. Several methods are defined to produce numerical and graphical summaries of these results (Section 3.3). A modular programming approach, which utilizes function closures, allows interested users to readily extend the package's functionality, by inserting functions for new test statistics and testing procedures (Section 3.4). Ongoing efforts are discussed in Section 4.

2 Methods

2.1 Multiple hypothesis testing framework

Hypothesis testing is concerned with using observed data to test hypotheses, i.e., make decisions, regarding properties of the unknown data generating distribution. Below, we discuss in turn the main ingredients of a multiple testing problem, namely: data, null and alternative hypotheses, test statistics, multiple testing procedure (MTP) to define rejection regions for the test statistics, Type I and Type II errors, and adjusted p-values. The crucial choice of a test statistics null distribution is addressed in Section 2.2. Specific proposals of MTPs are given in Sections 2.3 – 2.5.

Data. Let $X_1, \ldots, X_n$ be a *random sample* of n independent and identically distributed (i.i.d.) random variables, $X \sim P \in \mathcal{M}$, where the *data generating distribution* P is known to be an element of a particular *statistical model* $\mathcal{M}$ (i.e., a set of possibly non-parametric distributions).

Null and alternative hypotheses. In order to cover a broad class of testing problems, define M null hypotheses in terms of a collection of *submodels*, $\mathcal{M}(m) \subseteq \mathcal{M}$, $m = 1, \ldots, M$, for the data generating distribution P. The M *null hypotheses* are defined as $H_0(m) \equiv \mathrm{I}(P \in \mathcal{M}(m))$ and the corresponding *alternative hypotheses* as $H_1(m) \equiv \mathrm{I}(P \notin \mathcal{M}(m))$.
In many testing problems, the submodels concern *parameters*, i.e., functions of the data generating distribution P, $\Psi(P) = \psi = (\psi(m) : m = 1, \ldots, M)$, such as means, differences in means, correlations, and parameters in linear models, generalized linear models, survival models, time-series models, dose-response models, etc. One distinguishes between two types of testing problems: *one-sided tests*, where $H_0(m) = \mathrm{I}(\psi(m) \leq \psi_0(m))$, and *two-sided*

tests, where $H_0(m) = \mathrm{I}(\psi(m) = \psi_0(m))$. The hypothesized *null values*, $\psi_0(m)$, are frequently zero.
Let $\mathcal{H}_0 = \mathcal{H}_0(P) \equiv \{m : H_0(m) = 1\} = \{m : P \in \mathcal{M}(m)\}$ be the set of $h_0 \equiv |\mathcal{H}_0|$ true null hypotheses, where we note that $\mathcal{H}_0$ depends on the data generating distribution P. Let $\mathcal{H}_1 = \mathcal{H}_1(P) \equiv \mathcal{H}_0^c(P) = \{m : H_1(m) = 1\} = \{m : P \notin \mathcal{M}(m)\}$ be the set of $h_1 \equiv |\mathcal{H}_1| = M - h_0$ false null hypotheses, i.e., true positives. The goal of a multiple testing procedure is to accurately estimate the set $\mathcal{H}_0$, and thus its complement $\mathcal{H}_1$, while controlling probabilistically the number of false positives at a user-supplied level α.

Test statistics. A testing procedure is a data-driven rule for deciding whether or not to *reject* each of the M null hypotheses $H_0(m)$, i.e., declare that $H_0(m)$ is false (zero) and hence $P \notin \mathcal{M}(m)$. The decisions to reject or not the null hypotheses are based on an M–vector of *test statistics*, $T_n = (T_n(m) : m = 1, \ldots, M)$, that are functions of the data, $X_1, \ldots, X_n$. Denote the typically unknown (finite sample) *joint distribution* of the test statistics T_n by $Q_n = Q_n(P)$.
Single-parameter null hypotheses are commonly tested using *t-statistics*, i.e., standardized differences,

$$T_n(m) \equiv \frac{\text{Estimator} - \text{Null value}}{\text{Standard error}} = \sqrt{n}\frac{\psi_n(m) - \psi_0(m)}{\sigma_n(m)}. \qquad (1)$$

In general, the M–vector $\psi_n = (\psi_n(m) : m = 1, \ldots, M)$ denotes an asymptotically linear *estimator* of the parameter M–vector $\psi = (\psi(m) : m = 1, \ldots, M)$ and $(\sigma_n(m)/\sqrt{n} : m = 1, \ldots, M)$ denote consistent estimators of the *standard errors* of the components of ψ_n. For tests of means, one recovers the usual one-sample and two-sample t-statistics, where the $\psi_n(m)$ and $\sigma_n(m)$ are based on sample means and variances, respectively. In some settings, it may be appropriate to use (unstandardized) *difference statistics*, $T_n(m) \equiv \sqrt{n}(\psi_n(m) - \psi_0(m))$ Pollard and van der Laan (2004). Test statistics for other types of null hypotheses include F-statistics, χ^2-statistics, and likelihood ratio statistics.

Example: ALL microarray dataset. Suppose that, as in the analysis of the ALL dataset of Chiaretti et al. Chiaretti et al. (2004) (See accompanying vignette MTPALL), one is interested in identifying genes that are differentially expressed in two populations of ALL cancer patients, those with normal cytogenetic test status and those with abnormal test. The data consist of random J–vectors X, where the first M entries of X are microarray expression measures on M genes of interest and the last entry, $X(J)$, is an indicator

for cytogenetic test status (1 for normal, 0 for abnormal). Then, the parameter of interest is an M–vector of differences in mean expression measures in the two populations, $\psi(m) = E[X(m)|X(J) = 0] - E[X(m)|X(J) = 1]$, $m = 1, \ldots, M$. To identify genes with higher mean expression measures in the abnormal compared to the normal cytogenetics subjects, one can test the one-sided null hypotheses $H_0(m) = \mathrm{I}(\psi(m) \leq 0)$ vs. the alternative hypotheses $H_1(m) = \mathrm{I}(\psi(m) > 0)$, using two-sample Welch t-statistics

$$T_n(m) \equiv \frac{\bar{X}_{0,n_0}(m) - \bar{X}_{1,n_1}(m)}{\sqrt{\frac{\sigma^2_{0,n_0}(m)}{n_0} + \frac{\sigma^2_{1,n_1}(m)}{n_1}}}, \tag{2}$$

where n_k, $\bar{X}_{k,n_k}(m)$, and $\sigma^2_{k,n_k}(m)$ denote, respectively, the sample size, sample means, and sample variances, for patients with test status k, $k = 0, 1$. The null hypotheses are rejected, i.e., the corresponding genes are declared differentially expressed, for large values of the test statistics $T_n(m)$.

Multiple testing procedure. A *multiple testing procedure* (MTP) provides *rejection regions*, $\mathcal{C}_n(m)$, i.e., sets of values for each test statistic $T_n(m)$ that lead to the decision to reject the null hypothesis $H_0(m)$. In other words, a MTP produces a random (i.e., data-dependent) subset $\mathcal{R}_n$ of rejected hypotheses that estimates $\mathcal{H}_1$, the set of true positives,

$$\mathcal{R}_n = \mathcal{R}(T_n, Q_{0n}, \alpha) \equiv \{m : H_0(m) \text{ is rejected}\} = \{m : T_n(m) \in \mathcal{C}_n(m)\}, \tag{3}$$

where $\mathcal{C}_n(m) = \mathcal{C}(T_n, Q_{0n}, \alpha)(m)$, $m = 1, \ldots, M$, denote possibly random rejection regions. The long notation $\mathcal{R}(T_n, Q_{0n}, \alpha)$ and $\mathcal{C}(T_n, Q_{0n}, \alpha)(m)$ emphasizes that the MTP depends on: (i) the *data*, $X_1, \ldots, X_n$, through the M–vector of *test statistics*, $T_n = (T_n(m) : m = 1, \ldots, M)$; (ii) a test statistics *null distribution*, Q_{0n} (Section 2.2); and (iii) the *nominal level* α of the MTP, i.e., the desired upper bound for a suitably defined false positive rate.

Unless specified otherwise, it is assumed that large values of the test statistic $T_n(m)$ provide evidence against the corresponding null hypothesis $H_0(m)$, that is, we consider rejection regions of the form $\mathcal{C}_n(m) = (c_n(m), \infty)$, where $c_n(m)$ are to-be-determined *cut-offs*, or *critical values*.

Type I and Type II errors. In any testing situation, two types of errors can be committed: a *false positive*, or *Type I error*, is committed by rejecting a true null hypothesis, and a *false negative*, or *Type II error*, is committed when the test procedure fails to reject a false null hypothesis.

The situation can be summarized by Table 1, below, where the number of Type I errors is $V_n \equiv \sum_{m \in \mathcal{H}_0} \mathrm{I}(T_n(m) \in \mathcal{C}_n(m)) = |\mathcal{R}_n \cap \mathcal{H}_0|$ and the number of Type II errors is $U_n \equiv \sum_{m \in \mathcal{H}_1} \mathrm{I}(T_n(m) \notin \mathcal{C}_n(m)) = |\mathcal{R}_n^c \cap \mathcal{H}_1|$. Note that both U_n and V_n depend on the unknown data generating distribution P through the unknown set of true null hypotheses $\mathcal{H}_0 = \mathcal{H}_0(P)$. The numbers $h_0 = |\mathcal{H}_0|$ and $h_1 = |\mathcal{H}_1| = M - h_0$ of true and false null hypotheses are *unknown parameters*, the number of rejected hypotheses $R_n \equiv \sum_{m=1}^{M} \mathrm{I}(T_n(m) \in \mathcal{C}_n(m)) = |\mathcal{R}_n|$ is an *observable random variable*, and the entries in the body of the table, U_n, $h_1 - U_n$, V_n, and $h_0 - V_n$, are *unobservable random variables* (depending on P, through $\mathcal{H}_0(P)$).

Table 1: Type I and Type II errors in multiple hypothesis testing.

		Null hypotheses not rejected	Null hypotheses rejected	
Null hypotheses	true	$\lvert\mathcal{R}_n^c \cap \mathcal{H}_0\rvert$	$V_n = \lvert\mathcal{R}_n \cap \mathcal{H}_0\rvert$ (Type I errors)	$h_0 = \lvert\mathcal{H}_0\rvert$
	false	$U_n = \lvert\mathcal{R}_n^c \cap \mathcal{H}_1\rvert$ (Type II errors)	$\lvert\mathcal{R}_n \cap \mathcal{H}_1\rvert$	$h_1 = \lvert\mathcal{H}_1\rvert$
		$M - R_n$	$R_n = \lvert\mathcal{R}_n\rvert$	M

Ideally, one would like to simultaneously minimize both the chances of committing a Type I error and a Type II error. Unfortunately, this is not feasible and one seeks a *trade-off* between the two types of errors. A standard approach is to specify an acceptable level α for the Type I error rate and derive testing procedures, i.e., rejection regions, that aim to minimize the Type II error rate, i.e., maximize *power*, within the class of tests with Type I error rate at most α.

Type I error rates. When testing multiple hypotheses, there are many possible definitions for the Type I error rate (and power). Accordingly, we adopt a general definition of Type I error rates, as parameters, $\theta_n = \theta(F_{V_n,R_n})$, of the joint distribution F_{V_n,R_n} of the numbers of Type I errors V_n and rejected hypotheses R_n. Such a general representation covers the following commonly-used Type I error rates.

1. *Generalized family-wise error rate* (gFWER), or probability of at least

$(k+1)$ Type I errors, $k = 0, \ldots, (h_0 - 1)$,

$$gFWER(k) \equiv Pr(V_n > k) = 1 - F_{V_n}(k). \tag{4}$$

When $k = 0$, the gFWER is the usual *family-wise error rate*, FWER, controlled by the classical Bonferroni procedure.

2. *Per-comparison error rate* (PCER), or expected proportion of Type I errors among the M tests,

$$PCER \equiv \frac{1}{M} E[V_n] = \frac{1}{M} \int v dF_{V_n}(v). \tag{5}$$

3. *Tail probabilities for the proportion of false positives* (TPPFP) among the rejected hypotheses,

$$TPPFP(q) \equiv Pr(V_n/R_n > q) = 1 - F_{V_n/R_n}(q), \qquad q \in (0,1), \tag{6}$$

with the convention that $V_n/R_n \equiv 0$, if $R_n = 0$.

4. *False discovery rate* (FDR), or expected value of the proportion of false positives among the rejected hypotheses,

$$FDR \equiv E[V_n/R_n] = \int q dF_{V_n/R_n}(q), \tag{7}$$

again with the convention that $V_n/R_n \equiv 0$, if $R_n = 0$ Benjamini and Hochberg (1995).

Note that while the gFWER is a parameter of only the *marginal* distribution F_{V_n} for the number of Type I errors V_n (tail probability, or survivor function, for V_n), the TPPFP is a parameter of the *joint* distribution of (V_n, R_n) (tail probability, or survivor function, for V_n/R_n). Error rates based on the *proportion* of false positives (e.g., TPPFP and FDR) are especially appealing for the large-scale testing problems encountered in genomics, compared to error rates based on the *number* of false positives (e.g., gFWER), as they do not increase exponentially with the number of hypotheses. The above four error rates are part of the broad class of Type I error rates considered in Dudoit & van der Laan Dudoit and van der Laan (2004) and defined as tail probabilities $Pr(g(V_n, R_n) > q)$ and expected values $E[g(V_n, R_n)]$ for an arbitrary function $g(V_n, R_n)$ of the numbers of false positives V_n and rejected hypotheses R_n. The gFWER and TPPFP correspond to the special cases $g(V_n, R_n) = V_n$ and $g(V_n, R_n) = V_n/R_n$, respectively.

Adjusted p-values. The notion of p-value extends directly to multiple testing problems, as follows. Given a MTP, $\mathcal{R}_n = \mathcal{R}(T_n, Q_{0n}, \alpha)$, the *adjusted p-value*, $\widetilde{P}_{0n}(m) = \widetilde{P}(T_n, Q_{0n})(m)$, for null hypothesis $H_0(m)$, is defined as the smallest Type I error level α at which one would reject $H_0(m)$, that is,

$$\begin{aligned} \widetilde{P}_{0n}(m) &\equiv \inf\{\alpha \in [0,1] : \text{Reject } H_0(m) \text{ at MTP level } \alpha\} \\ &= \inf\{\alpha \in [0,1] : m \in \mathcal{R}_n\} \\ &= \inf\{\alpha \in [0,1] : T_n(m) \in \mathcal{C}_n(m)\}, \qquad m = 1, \ldots, M. \end{aligned} \tag{8}$$

As in single hypothesis tests, the smaller the adjusted p-value, the stronger the evidence against the corresponding null hypothesis. The main difference between unadjusted (i.e., for the test of a single hypothesis) and adjusted p-values is that the latter are defined in terms of the Type I error rate for the *entire* testing procedure, i.e., take into account the multiplicity of tests. For example, the adjusted p-values for the classical Bonferroni procedure for FWER control are given by $\widetilde{P}_{0n}(m) = \min(MP_{0n}(m), 1)$, where $P_{0n}(m)$ is the unadjusted p-value for the test of single hypothesis $H_0(m)$.
We now have two representations for a MTP, in terms of rejection regions for the test statistics and in terms of adjusted p-values

$$\mathcal{R}_n = \{m : T_n(m) \in \mathcal{C}_n(m)\} = \{m : \widetilde{P}_{0n}(m) \leq \alpha\}. \tag{9}$$

Again, as in the single hypothesis case, an advantage of reporting adjusted p-values, as opposed to only rejection or not of the hypotheses, is that the level α of the test does not need to be determined in advance, that is, results of the multiple testing procedure are provided for all α. Adjusted p-values are convenient and flexible summaries of the strength of the evidence against each null hypothesis, in terms of the Type I error rate for the entire MTP (gFWER, TPPFP, FDR, or any other suitably defined error rate).

Stepwise multiple testing procedures. One usually distinguishes between two main classes of multiple testing procedures, single-step and stepwise procedures. In *single-step procedures*, each null hypothesis is evaluated using a rejection region that is independent of the results of the tests of other hypotheses. Improvement in power, while preserving Type I error rate control, may be achieved by *stepwise procedures*, in which rejection of a particular null hypothesis depends on the outcome of the tests of other hypotheses. That is, the (single-step) test procedure is applied to a sequence of successively smaller nested random (i.e., data-dependent) subsets of null hypotheses, defined by the ordering of the test statistics (common cut-offs) or

unadjusted p-values (common-quantile cut-offs). In *step-down procedures*, the hypotheses corresponding to the *most significant* test statistics (i.e., largest absolute test statistics or smallest unadjusted p-values) are considered successively, with further tests depending on the outcome of earlier ones. As soon as one fails to reject a null hypothesis, no further hypotheses are rejected. In contrast, for *step-up procedures*, the hypotheses corresponding to the *least significant* test statistics are considered successively, again with further tests depending on the outcome of earlier ones. As soon as one hypothesis is rejected, all remaining more significant hypotheses are rejected.

Confidence regions. For the test of single-parameter null hypotheses and for any Type I error rate of the form $\theta(F_{V_n})$, Dudoit & van der Laan Dudoit and van der Laan (2004) and Pollard & van der Laan Pollard and van der Laan (2004) provide results on the correspondence between single-step MTPs and θ–specific *confidence regions*.

2.2 Test statistics null distribution

Test statistics null distribution. One of the main tasks in specifying a MTP is to derive rejection regions for the test statistics such that the Type I error rate is controlled at a desired level α, i.e., such that $\theta(F_{V_n,R_n}) \leq \alpha$, for finite sample control, or $\limsup_n \theta(F_{V_n,R_n}) \leq \alpha$, for asymptotic control. However, one is immediately faced with the problem that the *true distribution* $Q_n = Q_n(P)$ of the test statistics T_n is usually *unknown*, and hence, so are the distributions of the numbers of Type I errors, $V_n = \sum_{m \in \mathcal{H}_0} \mathrm{I}(T_n(m) \in \mathcal{C}_n(m))$, and rejected hypotheses, $R_n = \sum_{m=1}^{M} \mathrm{I}(T_n(m) \in \mathcal{C}_n(m))$. In practice, the test statistics *true distribution* $Q_n(P)$ is replaced by a *null distribution* Q_0 (or estimate thereof, Q_{0n}), in order to derive rejection regions, $\mathcal{C}(T_n, Q_0, \alpha)(m)$, and resulting adjusted p-values, $\widetilde{P}(T_n, Q_0)(m)$.
The choice of null distribution Q_0 is crucial, in order to ensure that (finite sample or asymptotic) control of the Type I error rate under the *assumed* null distribution Q_0 does indeed provide the required control under the *true* distribution $Q_n(P)$. For proper control, the null distribution Q_0 must be such that the Type I error rate under this assumed null distribution *dominates* the Type I error rate under the true distribution $Q_n(P)$. That is, one must have $\theta(F_{V_n,R_n}) \leq \theta(F_{V_0,R_0})$, for finite sample control, and $\limsup_n \theta(F_{V_n,R_n}) \leq \theta(F_{V_0,R_0})$, for asymptotic control, where V_0 and R_0 denote, respectively, the numbers of Type I errors and rejected hypotheses under the assumed null distribution Q_0.

For error rates $\theta(F_{V_n})$, defined as arbitrary parameters of the distribution of the number of Type I errors V_n, we propose as null distribution the asymptotic distribution Q_0 of the vector of null value shifted and scaled test statistics Dudoit and van der Laan (2004); Dudoit et al. (2004); van der Laan et al. (2004b,a); Pollard and van der Laan (2004):

$$Z_n(m) \equiv \sqrt{\min\left(1, \frac{\tau_0(m)}{Var[T_n(m)]}\right)}\Big(T_n(m) + \lambda_0(m) - E[T_n(m)]\Big). \quad (10)$$

For the test of single-parameter null hypotheses using t-statistics, the null values are $\lambda_0(m) = 0$ and $\tau_0(m) = 1$. For testing the equality of K population means using F-statistics, the null values are $\lambda_0(m) = 1$ and $\tau_0(m) = 2/(K-1)$, under the assumption of equal variances in the different populations. Dudoit et al. Dudoit et al. (2004) and van der Laan et al. van der Laan et al. (2004b) prove that this null distribution does indeed provide the desired asymptotic control of the Type I error rate $\theta(F_{V_n})$, for general data generating distributions (with arbitrary dependence structures among variables), null hypotheses (defined in terms of submodels for the data generating distribution), and test statistics (e.g., t-statistics, F-statistics).
For a broad class of testing problems, such as the test of single-parameter null hypotheses using t-statistics (as in Equation (1)), the null distribution Q_0 is an M–variate Gaussian distribution with mean vector zero and covariance matrix $\Sigma^*(P)$: $Q_0 = Q_0(P) \equiv N(0, \Sigma^*(P))$. For tests of means, where the parameter of interest is the M–dimensional mean vector $\Psi(P) = \psi = E[X]$, the estimator ψ_n is simply the M–vector of sample averages and $\Sigma^*(P)$ is the correlation matrix of $X \sim P$, $Cor[X]$. More generally, for an asymptotically linear estimator ψ_n, $\Sigma^*(P)$ is the correlation matrix of the vector influence curve (IC).
Note that the following important points distinguish our approach from existing approaches to Type I error rate control. Firstly, we are only concerned with Type I error control under the *true data generating distribution* P. The notions of weak and strong control (and associated subset pivotality, Westfall & Young Westfall and Young (1993), p. 42–43) are therefore irrelevant to our approach. Secondly, we propose a *null distribution for the test statistics* ($T_n \sim Q_0$), and not a data generating null distribution ($X \sim P_0 \in \cap_{m=1}^{M} \mathcal{M}(m)$). The latter practice does not necessarily provide proper Type I error control, as the test statistics' *assumed* null distribution $Q_n(P_0)$ and their *true* distribution $Q_n(P)$ may have different dependence structures (in the limit) for the true null hypotheses $\mathcal{H}_0$.

Bootstrap estimation of the test statistics null distribution. In practice, since the data generating distribution P is unknown, then so is the proposed null distribution $Q_0 = Q_0(P)$. Resampling procedures, such as bootstrap Procedure 1, below, may be used to conveniently obtain consistent estimators Q_{0n} of the null distribution Q_0 and of the resulting test statistic cut-offs and adjusted p-values.
Dudoit et al. Dudoit et al. (2004) and van der Laan et al. van der Laan et al. (2004b) show that single-step and step-down procedures based on consistent estimators of the null distribution Q_0 also provide asymptotic control of the Type I error rate. The reader is referred to these two articles and to Dudoit & van der Laan Dudoit and van der Laan (2004) for details on the choice of null distribution and various approaches for estimating this null distribution.
Having selected a suitable test statistics null distribution, there remains the main task of specifying rejection regions for each null hypothesis, i.e., cut-offs for each test statistic. Among the different approaches for defining rejection regions, we distinguish between single-step vs. stepwise procedures, and common cut-offs (i.e., the same cut-off c_0 is used for each test statistic) vs. common-quantile cut-offs (i.e., the cut-offs are the δ_0–quantiles of the marginal null distributions of the test statistics). The next three subsections discuss three main approaches for deriving rejection regions and corresponding adjusted p-values: single-step common-cut-off and common-quantile procedures for control of general Type I error rates $\theta(F_{V_n})$ (Section 2.3); step-down common-cut-off (maxT) and common-quantile (minP) procedures for control of the FWER (Section 2.4); augmentation procedures for control of the gFWER and TPPFP, based on an initial FWER-controlling procedure (Section 2.5).

Procedure 1 [Bootstrap estimation of the null distribution Q_0]

1. *Let $P_n^\star$ denote an estimator of the data generating distribution P. For the* non-parametric bootstrap*, $P_n^\star$ is simply the empirical distribution P_n, that is, samples of size n are drawn at random, with replacement from the observed data $X_1, \ldots, X_n$. For the* model-based bootstrap*, $P_n^\star$ is based on a model $\mathcal{M}$ for the data generating distribution P, such as the family of M–variate Gaussian distributions.*

2. *Generate B bootstrap samples, each consisting of n i.i.d. realizations of a random variable $X^\# \sim P_n^\star$.*

3. *For the bth bootstrap sample, $b = 1, \ldots, B$, compute an M–vector of test statistics, $T_n^\#(\cdot, b) = (T_n^\#(m, b) : m = 1, \ldots, M)$. Arrange these bootstrap statistics in an $M \times B$ matrix, $\mathbf{T}_n^\# = \big(T_n^\#(m, b)\big)$, with rows corresponding to the M null hypotheses and columns to the B bootstrap samples.*

4. *Compute row means, $E[{T_n}^\#(m, \cdot)]$, and row variances, $Var[{T_n}^\#(m, \cdot)]$, of the matrix $\mathbf{T}_n^\#$, to yield estimates of the true means $E[T_n(m)]$ and variances $Var[T_n(m)]$ of the test statistics, respectively.*

5. *Obtain an $M \times B$ matrix, $\mathbf{Z}_n^\# = \big(Z_n^\#(m, b)\big)$, of null value shifted and scaled bootstrap statistics $Z_n^\#(m, b)$, by row-shifting and scaling the matrix $\mathbf{T}_n^\#$ using the bootstrap estimates of $E[T_n(m)]$ and $Var[T_n(m)]$ and the user-supplied null values $\lambda_0(m)$ and $\tau_0(m)$. That is, compute*

$$Z_n^\#(m, b) \equiv \sqrt{\min\left(1, \frac{\tau_0(m)}{Var[{T_n}^\#(m, \cdot)]}\right)} \times \Big(T_n^\#(m, b) + \lambda_0(m) - E[{T_n}^\#(m, \cdot)]\Big). \tag{11}$$

6. *The bootstrap estimate Q_{0n} of the null distribution Q_0 is the empirical distribution of the B columns $Z_n^\#(\cdot, b)$ of matrix $\mathbf{Z}_n^\#$.*

2.3 Single-step procedures for control of general Type I error rates $\theta(F_{V_n})$

Dudoit et al. Dudoit et al. (2004) and Pollard & van der Laan Pollard and van der Laan (2004) propose single-step common-cut-off and common-quantile procedures for controlling arbitrary parameters $\theta(F_{V_n})$ of the distribution of the number of Type I errors. The main idea is to substitute control of the parameter $\theta(F_{V_n})$, for the *unknown, true distribution* F_{V_n} of the number of Type I errors, by control of the corresponding parameter $\theta(F_{R_0})$, for the *known, null distribution* F_{R_0} of the number of rejected hypotheses. That is, consider single-step procedures of the form $\mathcal{R}_n \equiv \{m : T_n(m) > c_n(m)\}$, where the cut-offs $c_n(m)$ are chosen so that $\theta(F_{R_0}) \leq \alpha$, for $R_0 \equiv \sum_{m=1}^{M} \mathrm{I}(Z(m) > c_n(m))$ and $Z \sim Q_0$. Among the class of MTPs that satisfy $\theta(F_{R_0}) \leq \alpha$, Dudoit et al. Dudoit et al. (2004) and Pollard & van der Laan Pollard and van der Laan (2004) propose two procedures, based on common cut-offs and common-quantile cut-offs, respectively. The procedures are summarized below and the reader is referred to the articles for proofs and details on the derivation of cut-offs and adjusted p-values.

Single-step common-cut-off procedure. The set of rejected hypotheses for the *θ–controlling single-step common-cut-off procedure* is of the form $\mathcal{R}_n \equiv \{m : T_n(m) > c_0\}$, where the common cut-off c_0 is the *smallest* (i.e., least conservative) value for which $\theta(F_{R_0}) \leq \alpha$.
For $gFWER(k)$ control (special case $\theta(F_{V_n}) = 1 - F_{V_n}(k)$), the procedure is based on the $(k+1)$*st ordered test statistic*. Specifically, the adjusted p-values are given by

$$\widetilde{p}_{0n}(m) = Pr_{Q_0}\left(Z^{\circ}(k+1) \geq t_n(m)\right), \qquad m = 1, \ldots, M, \tag{12}$$

where $Z^{\circ}(m)$ denotes the mth ordered component of $Z = (Z(m) : m = 1, \ldots, M) \sim Q_0$, so that $Z^{\circ}(1) \geq \ldots \geq Z^{\circ}(M)$. For FWER control ($k = 0$), the procedure reduces to the *single-step maxT procedure*, based on the *maximum test statistic*, $Z^{\circ}(1)$.

Single-step common-quantile procedure. The set of rejected hypotheses for the *θ–controlling single-step common-quantile procedure* is of the form $\mathcal{R}_n \equiv \{m : T_n(m) > c_0(m)\}$, where $c_0(m) = Q_{0,m}^{-1}(\delta_0)$ is the δ_0–quantile of the marginal null distribution $Q_{0,m}$ of the mth test statistic, i.e., the smallest value c such that $Q_{0,m}(c) = Pr_{Q_0}(Z(m) \leq c) \geq \delta_0$ for $Z \sim Q_0$. Here, δ_0 is chosen as the *smallest* (i.e., least conservative) value for which $\theta(F_{R_0}) \leq \alpha$.

For $gFWER(k)$ control, the procedure is based on the $(k+1)st$ *ordered unadjusted p-value*. Specifically, let $\bar{Q}_{0,m} \equiv 1 - Q_{0,m}$ denote the survivor functions for the marginal null distributions $Q_{0,m}$ and define unadjusted p-values $P_0(m) \equiv \bar{Q}_{0,m}(Z(m))$ and $P_{0n}(m) \equiv \bar{Q}_{0,m}(T_n(m))$, for $Z \sim Q_0$ and $T_n \sim Q_n$, respectively. Then, the adjusted p-values for the common-quantile procedure are given by

$$\widetilde{p}_{0n}(m) = Pr_{Q_0}\left(P_0^\circ(k+1) \leq p_{0n}(m)\right), \qquad m = 1, \ldots, M, \tag{13}$$

where $P_0^\circ(m)$ denotes the mth ordered component of the M–vector of unadjusted p-values $(P_0(m) : m = 1, \ldots, M)$, so that $P_0^\circ(1) \leq \ldots \leq P_0^\circ(M)$. For FWER control ($k = 0$), one recovers the *single-step minP procedure*, based on the *minimum unadjusted p-value*, $P_0^\circ(1)$.

2.4 Step-down procedures for control of the family-wise error rate

van der Laan et al. van der Laan et al. (2004b) propose step-down common-cut-off (maxT) and common-quantile (minP) procedures for controlling the family-wise error rate, FWER. These procedures are similar in spirit to their single-step counterparts in Section 2.3 (special case $\theta(F_{V_n}) = 1 - F_{V_n}(0)$), with the important step-down distinction that hypotheses are considered successively, from most significant to least significant, with further tests depending on the outcome of earlier ones. That is, the test procedure is applied to a sequence of successively smaller nested random (i.e., data-dependent) subsets of null hypotheses, defined by the ordering of the test statistics (common cut-offs) or unadjusted p-values (common-quantile cut-offs).

Step-down common-cut-off (maxT) procedure. Rather than being based solely on the distribution of the maximum test statistic over all M hypotheses, the step-down common cut-offs and corresponding adjusted p-values are based on the distributions of maxima of test statistics over successively smaller nested random subsets of null hypotheses. Specifically, let $O_n(m)$ denote the indices for the ordered test statistics $T_n(m)$, so that $T_n(O_n(1)) \geq \ldots \geq T_n(O_n(M))$. The step-down common-cut-off procedure is then based on the distributions of maxima of test statistics over the nested subsets of ordered hypotheses $\overline{\mathcal{O}}_n(h) \equiv \{O_n(h), \ldots, O_n(M)\}$. The adjusted p-values for the *step-down maxT procedure* are given by

$$\widetilde{p}_{0n}(o_n(m)) = \max_{h=1,\ldots,m} \left\{ Pr_{Q_0}\left(\max_{l \in \bar{l}_n(h)} Z(l) \geq t_n(o_n(h)) \right) \right\}, \tag{14}$$

where $Z = (Z(m) : m = 1, \ldots, M) \sim Q_0$. Taking maxima of the probabilities over $h \in \{1, \ldots, m\}$ enforces monotonicity of the adjusted p-values and ensures that the procedure is indeed step-down, that is, one can only reject a particular hypothesis provided all hypotheses with more significant (i.e., larger) test statistics were rejected beforehand.

Step-down common-quantile (minP) procedure. Likewise, the step-down common-quantile cut-offs and corresponding adjusted p-values are based on the distributions of minima of unadjusted p-values over successively smaller nested random subsets of null hypotheses. Specifically, let $O_n(m)$ denote the indices for the ordered unadjusted p-values $P_{0n}(m)$, so that $P_{0n}(O_n(1)) \leq \ldots \leq P_{0n}(O_n(M))$. The step-down common-quantile procedure is then based on the distributions of minima of unadjusted p-values over the nested subsets of ordered hypotheses $\overline{\mathcal{O}}_n(h) \equiv \{O_n(h), \ldots, O_n(M)\}$. The adjusted p-values for the *step-down minP procedure* are given by

$$\widetilde{p}_{0n}(o_n(m)) = \max_{h=1,\ldots,m} \left\{ Pr_{Q_0} \left(\min_{l \in \bar{\iota}_n(h)} P_0(l) \leq p_{0n}(o_n(h)) \right) \right\}, \qquad (15)$$

where $P_0(m) = \bar{Q}_{0,m}(Z(m))$ and $Z = (Z(m) : m = 1, \ldots, M) \sim Q_0$.

2.5 Augmentation multiple testing procedures

Dudoit & van der Laan Dudoit and van der Laan (2004) and van der Laan et al. van der Laan et al. (2004a) discuss *augmentation multiple testing procedures* (AMTP), obtained by adding suitably chosen null hypotheses to the set of null hypotheses already rejected by an initial MTP. Specifically, given *any* initial procedure controlling the generalized family-wise error rate, augmentation procedures are derived for controlling Type I error rates defined as tail probabilities and expected values for arbitrary functions $g(V_n, R_n)$ of the numbers of Type I errors and rejected hypotheses (e.g., proportion $g(V_n, R_n) = V_n/R_n$ of false positives among the rejected hypotheses). Adjusted p-values for the AMTP are shown to be simply shifted versions of the adjusted p-values of the original MTP. The important practical implication of these results is that *any* FWER-controlling MTP and its corresponding adjusted p-values, provide, without additional work, multiple testing procedures controlling a broad class of Type I error rates and their adjusted p-values. One can therefore build on the large pool of available FWER-controlling procedures, such as the single-step and step-down maxT and minP procedures discussed in Sections 2.3 and 2.4, above.

Augmentation procedures for controlling tail probabilities of the number (gFWER) and proportion (TPPFP) of false positives, based on an initial FWER-controlling procedure, are treated in detail in van der Laan et al. van der Laan et al. (2004a) and are summarized below. The gFWER and TPPFP correspond to the special cases $g(V_n, R_n) = V_n$ and $g(V_n, R_n) = V_n/R_n$, respectively. Denote the adjusted p-values for the initial FWER-controlling procedure by $\widetilde{P}_{0n}(m)$. Order the M null hypotheses according to these p-values, from smallest to largest, that is, define indices $O_n(m)$, so that $\widetilde{P}_{0n}(O_n(1)) \leq \ldots \leq \widetilde{P}_{0n}(O_n(M))$. Then, for a nominal level α test, the initial FWER-controlling procedure rejects the R_n null hypotheses

$$\mathcal{R}_n \equiv \{m : \widetilde{P}_{0n}(m) \leq \alpha\}. \tag{16}$$

Augmentation procedure for controlling the gFWER. For control of $gFWER(k)$ at level α, given an initial FWER-controlling procedure, reject the R_n hypotheses specified by this MTP, as well as the next $A_n = \min\{k, M - R_n\}$ most significant null hypotheses. The adjusted p-values $\widetilde{P}^+_{0n}(O_n(m))$ for the new gFWER-controlling AMTP are simply k–shifted versions of the adjusted p-values of the initial FWER-controlling MTP:

$$\widetilde{P}^+_{0n}(O_n(m)) = \begin{cases} 0, & \text{if } m = 1, \ldots, k, \\ \widetilde{P}_{0n}(O_n(m-k)), & \text{if } m = k+1, \ldots, M. \end{cases} \tag{17}$$

That is, the first k adjusted p-values are set to zero and the remaining p-values are the adjusted p-values of the FWER-controlling MTP shifted by k. The AMTP thus guarantees at least k rejected hypotheses.

Augmentation procedure for controlling the TPPFP. For control of $TPPFP(q)$ at level α, given an initial FWER-controlling procedure, reject the R_n hypotheses specified by this MTP, as well as the next A_n most significant null hypotheses,

$$\begin{aligned} A_n &= \max\left\{m \in \{0, \ldots, M - R_n\} : \frac{m}{m + R_n} \leq q\right\} \\ &= \min\left\{\left\lfloor \frac{qR_n}{1-q} \right\rfloor, M - R_n\right\}, \end{aligned} \tag{18}$$

where the *floor* $\lfloor x \rfloor$ denotes the greatest integer less than or equal to x, i.e., $\lfloor x \rfloor \leq x < \lfloor x \rfloor + 1$. That is, keep rejecting null hypotheses until the ratio of additional rejections to the total number of rejections reaches the allowed proportion q of false positives. The adjusted p-values $\widetilde{P}^+_{0n}(O_n(m))$ for the

new TPPFP-controlling AMTP are simply shifted versions of the adjusted p-values of the initial FWER-controlling MTP, that is,

$$\widetilde{P}_{0n}^{+}(O_n(m)) = \widetilde{P}_{0n}(O_n(\lceil (1-q)m \rceil)), \qquad m = 1, \ldots, M, \tag{19}$$

where the *ceiling* $\lceil x \rceil$ denotes the least integer greater than or equal to x, i.e., $\lceil x \rceil - 1 < x \leq \lceil x \rceil$.

FDR-controlling procedures. Given any TPPFP-controlling procedure, van der Laan et al. van der Laan et al. (2004a) derive two simple (conservative) FDR-controlling procedures. The more general and conservative procedure controls the FDR at nominal level α, by controlling $TPPFP(\alpha/2)$ at level $\alpha/2$. The less conservative procedure controls the FDR at nominal level α, by controlling $TPPFP(1-\sqrt{1-\alpha})$ at level $1-\sqrt{1-\alpha}$. In what follows, we refer to these two MTPs as "conservative" and "restricted", respectively. The reader is referred to the original article for details and proofs of FDR control (Section 2.4, Theorem 3).

3 Software implementation: *multtest* package

3.1 Overview

The MTPs proposed in Sections 2.3 – 2.5 are implemented in the latest version of the Bioconductor R package *multtest* (version 1.5.0, Bioconductor release 1.5). New features include: expanded class of tests (e.g., for regression parameters in linear models and in Cox proportional hazards models); control of a wider selection of Type I error rates (e.g., gFWER, TPPFP, FDR); bootstrap estimation of the test statistics null distribution; augmentation multiple testing procedures; confidence regions for the parameter vector of interest. Because of their general applicability and novelty, we focus in this section on MTPs that utilize a bootstrap estimated test statistics null distribution and that are available through the package's main user-level function: `MTP`. Note that for many testing problems, MTPs based on permutation (rather than bootstrap) estimated null distributions are also available in the present and earlier versions of *multtest*. In particular, permutation-based step-down maxT and minP FWER-controlling MTPs are implemented in the functions `mt.maxT` and `mt.minP`, respectively, and can also be applied directly through a call to the `MTP` function.
We stress that *all* the bootstrap-based MTPs implemented in *multtest* can be performed using the main user-level function `MTP`. Most users will therefore

only need to be familiar with this function. Other functions are provided primarily for the benefit of more advanced users, interested in extending the package's functionality (Section 3.4). For greater detail on *multtest* functions, the reader is referred to the package documentation, in the form of help files, e.g., `? MTP`, and vignettes, e.g., `openVignette("multtest")`. One needs to specify the following main ingredients when applying a MTP: the *data*, $X_1, \ldots, X_n$; suitably defined *test statistics*, T_n, for each of the null hypotheses under consideration (e.g., one-sample t-statistics, robust rank-based F-statistics, t-statistics for regression coefficients in Cox proportional hazards model); a choice of *Type I error rate*, $\theta(F_{V_n,R_n})$, providing an appropriate measure of false positives for the particular testing problem (e.g., $TPPFP(0.10)$); a proper *joint null distribution*, Q_0 (or estimate thereof, Q_{0n}), for the test statistics (e.g., bootstrap null distribution as in Procedure 1); given the previously defined components, a *multiple testing procedure*, $\mathcal{R}_n = \mathcal{R}(T_n, Q_{0n}, \alpha)$, for controlling the error rate $\theta(F_{V_n,R_n})$ at a target level α. Accordingly, the *multtest* package has adopted a modular and extensible approach to the implementation of MTPs, with the following four main types of functions.

- Functions for computing the *test statistics*, T_n. These are internal functions (e.g., `meanX`, `coxY`), i.e., functions that are generally not called directly by the user. As shown in Section 3.2, below, the type of test statistic is specified by the `test` argument of the main user-level function `MTP`. Advanced users, interested in extending the class of tests available in *multtest*, can simply add their own test statistic functions to the existing library of such internal functions (see Section 3.4, below, for a brief discussion of the closure approach for specifying test statistics).

- Functions for obtaining the *test statistics null distribution*, Q_0, or an estimate thereof, Q_{0n}. The main function currently available is the internal function `boot.resample`, implementing the non-parametric version of bootstrap Procedure 1 (Section 2.2).

- Functions for implementing the *multiple testing procedure*, $\mathcal{R}(T_n, Q_{0n}, \alpha)$, i.e., for deriving rejection regions, confidence regions, and adjusted p-values. The main function is the user-level wrapper function `MTP`, which implements the single-step and step-down maxT and minP procedures for FWER control (Sections 2.3 and 2.4). The functions `fwer2gfwer`, `fwer2tppfp`, and `fwer2fdr` implement, respectively, gFWER-, TPPFP-, and FDR-controlling augmentation multiple testing proce-

dures, based on adjusted p-values from *any* FWER-controlling procedure, and can be called via the `typeone` argument to `MTP` (Section 2.5).

- Functions for *numerical and graphical summaries* of a MTP. As described in Section 3.3, below, a number of summary methods are available to operate on objects of class *MTP*, output from the main `MTP` function.

3.2 Resampling-based multiple testing procedures: `MTP` function

The main user-level function for resampling-based multiple testing is `MTP`. Its input/output and usage are described next.

```
> library(Biobase)
> library(multtest)

> args(MTP)

function (X, W = NULL, Y = NULL, Z = NULL, Z.incl = NULL, Z.test = NULL,
    na.rm = TRUE, test = "t.twosamp.unequalvar", robust = FALSE,
    standardize = TRUE, alternative = "two.sided", psi0 = 0,
    typeone = "fwer", k = 0, q = 0.1, fdr.method = "conservative",
    alpha = 0.05, nulldist = "boot", B = 1000, method = "ss.maxT",
    get.cr = FALSE, get.cutoff = FALSE, get.adjp = TRUE, keep.nulldist = FALSE,
    seed = NULL)
NULL
```

INPUT.

Data. The data, `X`, consist of a J–dimensional random vector, observed on each of n sampling units (patients, cell lines, mice, etc). These data can be stored in a $J \times n$ *matrix*, *data.frame*, or *exprs* slot of an object of class *exprSet*. In some settings, a J–vector of weights may be associated with each observation, and stored in a $J \times n$ weight matrix, `W` (or an n–vector `W`, if the weights are the same for each of the J variables). One may also observe a possibly censored continuous or polychotomous outcome, `Y`, for each sampling unit, as obtained, for example, from the *phenoData* slot of an object of class *exprSet*. In some studies, L additional covariates may be measured on each sampling unit and stored in `Z`, an $n \times L$ *matrix* or *data.frame*. When the tests

concern parameters in regression models with covariates from `Z` (e.g., values `lm.XvsZ`, `lm.YvsXZ`, and `coxph.YvsXZ`, for the argument `test`, described below), the arguments `Z.incl` and `Z.test` specify, respectively, which covariates (i.e., which columns of `Z`, including `Z.test`) should be included in the model and which regression parameter is to be tested (only when `test="lm.XvsZ"`). The covariates can be specified either by a numeric column index or character string. If `X` is an instance of the class *exprSet*, `Y` can be a column index or character string referring to the variable in the *data.frame* `pData(X)` to use as outcome. Likewise, `Z.incl` and `Z.test` can be column indices or character strings referring to the variables in `pData(X)` to use as covariates. The data components (`X`, `W`, `Y`, `Z`, `Z.incl`, and `Z.test`) are the first six arguments to the `MTP` function. Only `X` is a required argument; the others are by default `NULL`. The argument `na.rm` allows one to control the treatment of "Not Available" or `NA` values. It is set to `TRUE`, by default, so that an observation with a missing value in any of the data objects' jth component ($j = 1, \ldots, J$) is excluded from computation of any of the relevant test statistics.

Test statistics.

The test statistics should be chosen based on the parameter of interest (e.g., location, scale, or regression parameters) and the hypotheses one wishes to test. In the current implementation of *multtest*, the following test statistics are available through the argument `test`, with default value `t.twosamp.unequalvar`, for the two-sample Welch t-statistic.

- `t.onesamp`: One-sample t-statistic for tests of means.
- `t.twosamp.equalvar`: Equal variance two-sample t-statistic for tests of differences in means.
- `t.twosamp.unequalvar`: Unequal variance two-sample t-statistic for tests of differences in means (also known as two-sample Welch t-statistic).
- `t.pair`: Two-sample paired t-statistic for tests of differences in means.
- `f`: Multi-sample F-statistic for tests of equality of population means.
- `f.block`: Multi-sample F-statistic for tests of equality of population means in a block design.

- `lm.XvsZ`: t-statistic for tests of regression coefficients for variable `Z.test` in linear models each with outcome `X[j,]` ($j = 1, \ldots, J$), and possibly additional covariates `Z.incl` from the *matrix* `Z` (in the case of no covariates, one recovers the one-sample t-statistic, `t.onesamp`).
- `lm.YvsXZ`: t-statistic for tests of regression coefficients in linear models with outcome `Y` and each `X[j,]` ($j = 1, \ldots, J$) as covariate of interest, with possibly other covariates `Z.incl` from the *matrix* `Z`.
- `coxph.YvsXZ`: t-statistic for tests of regression coefficients in Cox proportional hazards survival models with outcome `Y` and each `X[j,]` ($j = 1, \ldots, J$) as covariate of interest, with possibly other covariates `Z.incl` from the *matrix* `Z`.

Robust, *rank-based* versions of the above test statistics can be specified by setting the argument `robust` to `TRUE` (the default value is `FALSE`). Consideration should be given to whether *standardized* (Equation (1)) or *unstandardized* difference statistics are most appropriate (see Pollard & van der Laan Pollard and van der Laan (2004) for a comparison). Both options are available through the argument `standardize`, by default `TRUE`. The type of alternative hypotheses is specified via the `alternative` argument: default value of `two.sided`, for two-sided test, and values of `less` or `greater`, for one-sided tests. The (common) null value for the parameters of interest is specified through the `psi0` argument, by default zero.

Type I error rate. The `MTP` function controls by default the family-wise error rate (FWER), or chance of at least one false positive (argument `typeone="fwer"`). Augmentation procedures (Section 2.5), controlling other Type I error rates such as the gFWER, TPPFP, and FDR, can be specified through the argument `typeone`. Related arguments include `k` and `q`, for the allowed number and proportion of false positives for control of $gFWER(k)$ and $TPPFP(q)$, respectively, and `fdr.method`, for the type of TPPFP-based FDR-controlling procedure (i.e., `"conservative"` or `"restricted"` methods). The nominal level of the test is determined by the argument `alpha`, by default 0.05. Testing can be performed for a range of nominal Type I error rates by specifying a vector of levels `alpha`.

Test statistics null distribution. In the current implementation of `MTP`, the test statistics null distribution is estimated by default using the non-

parametric version of bootstrap Procedure 1 (argument `nulldist="boot"`). The bootstrap procedure is implemented in the internal function `boot.resample`, which calls C to compute test statistics for each bootstrap sample. The values of the shift (λ_0) and scale (τ_0) parameters are determined by the type of test statistics (e.g., $\lambda_0 = 0$ and $\tau_0 = 1$ for t-statistics). Permutation null distributions are also available via `nulldist="perm"`. The number of resampling steps is specified by the argument `B`, by default 1,000.

Multiple testing procedures. Several methods for controlling the chosen Type I error rate are available in *multtest*.

- *FWER-controlling procedures.* For FWER control, the `MTP` function implements the single-step and step-down (common-cut-off) maxT and (common-quantile) minP MTPs, described in Sections 2.3 and 2.4, and specified through the argument `method` (internal functions `ss.maxT`, `ss.minP`, `sd.maxT`, and `sd.minP`). The default MTP is the single-step maxT procedure (`method="ss.maxT"`), since it requires the least computation.
- *gFWER-, TPPFP-, and FDR-controlling augmentation procedures.* As discussed in Section 2.5, any FWER-controlling MTP can be trivially augmented to control additional Type I error rates, such as the gFWER and TPPFP. Two FDR-controlling procedures can then be derived from the TPPFP-controlling AMTP. The AMTPs are implemented in the functions `fwer2gfwer`, `fwer2tppfp`, and `fwer2fdr`, that take FWER adjusted p-values as input and return augmentation adjusted p-values for control of the gFWER, TPPFP, and FDR, respectively. Note that the aforementioned AMTPs can be applied directly via the `typeone` argument of the main function `MTP`.

Output control. Various arguments are available to control output, i.e., specify which combination of the following quantities should be returned: confidence regions (argument `get.cr`); cut-offs for the test statistics (argument `get.cutoff`); adjusted p-values (argument `get.adjp`); test statistics null distribution (argument `keep.nulldist`). Note that parameter estimates and confidence regions only apply to the test of single-parameter null hypotheses (i.e., not the F-tests). In addition, in the current implementation of `MTP`, parameter confidence regions and test statistic cut-offs are only provided when `typeone="fwer"`, so that

`get.cr` and `get.cutoff` should be set to `FALSE` when using the error rates gFWER, TPPFP, or FDR.

Note that the *multtest* package also provides several simple, marginal FWER-controlling MTPs, such as the Bonferroni, Holm Holm (1979), Hochberg Hochberg (1988), and Šidák Šidák (1967) procedures, and FDR-controlling MTPs, such as the Benjamini & Hochberg Benjamini and Hochberg (1995) and Benjamini & Yekutieli Benjamini and Yekutieli (2001) procedures. These procedures are available through the `mt.rawp2adjp` function, which takes a vector of unadjusted p-values as input and returns the corresponding adjusted p-values.

OUTPUT.

The S4 class/method object-oriented programming approach was adopted to summarize the results of a MTP (Section 3.4). Specifically, the output of the `MTP` function is an instance of the *class MTP*. A brief description of the class and associated methods is given next. Please consult the documentation for details, e.g., using `class ? MTP` and `methods ? MTP`.

```
> slotNames("MTP")

 [1] "statistic" "estimate"  "sampsize"  "rawp"      "adjp"      "conf.reg"
 [7] "cutoff"    "reject"    "nulldist"  "call"      "seed"
```

`statistic`: The numeric M–vector of test statistics, specified by the values of the `MTP` arguments `test`, `robust`, `standardize`, and `psi0`. In many testing problems, $M = J =$ `nrow(X)`.

`estimate`: For the test of single-parameter null hypotheses using t-statistics (i.e., not the F-tests), the numeric M–vector of estimated parameters.

`sampsize`: The sample size, i.e., $n =$ `ncol(X)`.

`rawp`: The numeric M–vector of unadjusted p-values.

`adjp`: The numeric M–vector of adjusted p-values (computed only if the `get.adjp` argument is `TRUE`).

`conf.reg`: For the test of single-parameter null hypotheses using t-statistics (i.e., not the F-tests), the numeric $M \times 2 \times$ `length(alpha)` *array* of lower and upper simultaneous confidence limits for the parameter vector, for each value of the nominal Type I error rate `alpha` (computed only if the `get.cr` argument is `TRUE`).

`cutoff`: The numeric $M\times$ `length(alpha)` *matrix* of cut-offs for the test statistics, for each value of the nominal Type I error rate `alpha` (computed only if the `get.cutoff` argument is `TRUE`).

`reject`: The $M\times$ `length(alpha)` *matrix* of rejection indicators (`TRUE` for a rejected null hypothesis), for each value of the nominal Type I error rate `alpha`.

`nulldist`: The numeric $M \times B$ *matrix* for the estimated test statistics null distribution (returned only if `keep.nulldist=TRUE`; option not currently available for permutation null distribution, i.e., `nulldist="perm"`). By default (i.e., for `nulldist="boot"`), the entries of `nulldist` are the null value shifted and scaled bootstrap test statistics, as defined by Procedure 1.

`call`: The call to the function `MTP`.

`seed`: An integer for specifying the state of the random number generator used to create the resampled datasets. The seed can be reused for reproducibility in a repeat call to `MTP`. This argument is currently used only for the bootstrap null distribution (i.e., for `nulldist="boot"`). See `? set.seed` for details.

3.3 Numerical and graphical summaries

The following *methods* are defined to operate on *MTP* instances and summarize the results of a MTP.

`print`: The `print` method returns a description of an object of class *MTP*, including the sample size n, the number M of tested hypotheses, the type of test performed (value of argument `test`), the Type I error rate (value of argument `typeone`), the nominal level of the test (value of argument `alpha`), the name of the MTP (value of argument `method`), the call to the function `MTP`. In addition, this method produces a table with the class, mode, length, and dimension of each slot of the *MTP* instance.

`summary`: The `summary` method provides numerical summaries of the results of a MTP and returns a list with the following three components.

- `rejections`: A *data.frame* with the number(s) of rejected hypotheses for the nominal Type I error rate(s) specified by the

`alpha` argument of the function `MTP` (`NULL` values are returned if all three arguments `get.cr`, `get.cutoff`, and `get.adjp` are `FALSE`).

- `index`: A numeric M–vector of indices for ordering the hypotheses according to first `adjp`, then `rawp`, and finally the absolute value of `statistic` (not printed in the summary).
- `summaries`: When applicable (i.e., when the corresponding quantities are returned by `MTP`), a table with six number summaries of the distributions of the adjusted p-values, unadjusted p-values, test statistics, and parameter estimates.

`plot`: The `plot` method produces the following graphical summaries of the results of a MTP. The type of display may be specified via the `which` argument.

1. Scatterplot of number of rejected hypotheses vs. nominal Type I error rate.
2. Plot of ordered adjusted p-values; can be viewed as a plot of Type I error rate vs. number of rejected hypotheses.
3. Scatterplot of adjusted p-values vs. test statistics (also known as "volcano plot").
4. Plot of unordered adjusted p-values.
5. Plot of confidence regions for user-specified parameters, by default the 10 parameters corresponding to the smallest adjusted p-values (argument `top`).
6. Plot of test statistics and corresponding cut-offs (for each value of `alpha`) for user-specified hypotheses, by default the 10 hypotheses corresponding to the smallest adjusted p-values (argument `top`).

The argument `logscale` (by default equal to `FALSE`) allows one to use the negative decimal logarithms of the adjusted p-values in the second, third, and fourth graphical displays. Note that some of these plots are implemented in the older function `mt.plot`.

`[`: Subsetting method, which operates selectively on each slot of an *MTP* instance to retain only the data related to the specified hypotheses.

`as.list`: Converts an object of class *MTP* to an object of class *list*, with an entry for each slot.

3.4 Software design

The following features of the programming approach employed in *multtest* may be of interest to users, especially those interested in extending the functionality of the package.

Function closures. The use of *function closures*, in the style of the *genefilter* package, allows uniform data input for all MTPs and facilitates the extension of the package's functionality by adding, for example, new types of test statistics. Specifically, for each value of the `MTP` argument `test`, a closure is defined which consists of a function for computing the test statistic (with only two arguments, a data vector `x` and a corresponding weight vector `w`, with default value of `NULL`) and its enclosing environment, with bindings for relevant additional arguments, such as null values `psi0`, outcomes `Y`, and covariates `Z`. Thus, new test statistics can be added to *multtest* by simply defining a new closure and adding a corresponding value for the `test` argument to `MTP` (existing internal test statistic functions are located in the file `R/statistics.R`).

Class/method object-oriented programming. Like many other Bioconductor packages, *multtest* has adopted the *S4 class/method object-oriented programming approach* of Chambers Chambers (1998). In particular, a new class, *MTP*, is defined to represent the results of multiple testing procedures, as implemented in the main `MTP` function. As discussed above, in Section 3.3, several methods are provided to operate on instances of this class.

Calls to C. Because resampling procedures, such as the non-parametric bootstrap implemented in *multtest*, are computationally intensive, care must be taken to ensure that the resampling steps are not prohibitively slow. The use of closures for the test statistics, however, prevents writing the entire program in C. In the current implementation, we have chosen to define the closure and compute the observed test statistics in R, and then call C (using the R random number generator) to apply the closure to each bootstrap resampled dataset. This approach puts the for loops over bootstrap samples (B) and hypotheses (M) in the C environment, thus speeding up this computationally expensive part of the program. Further optimization for speed may be investigated for future releases.

4 Discussion

The *multtest* package implements a broad range of resampling-based multiple testing procedures. Ongoing efforts are as follows.

1. Extending the class of available tests, by adding test statistic closures for tests of correlations, quantiles, and parameters in generalized linear models (e.g., logistic regression).

2. Extending the class of resampling-based estimators for the test statistics null distribution (e.g., parametric bootstrap, Bayesian bootstrap). A closure approach may be considered for this purpose.

3. Providing parameter confidence regions and test statistic cut-offs for other Type I error rates than the FWER.

4. Implementing the new augmentation multiple testing procedures proposed in Dudoit & van der Laan Dudoit and van der Laan (2004) for controlling tail probabilities $Pr(g(V_n, R_n) > q)$ for an arbitrary function $g(V_n, R_n)$ of the numbers of false positives V_n and rejected hypotheses R_n.

5. Providing a formula interface for a symbolic description of the tests to be performed (cf. model specification in `lm`).

6. Providing an `update` method for objects of class *MTP*. This would allow reusing available estimates of the null distribution to implement different MTPs for a given Type I error rate and to control different Type I error rates.

7. Extending the *MTP* class to keep track of results for several MTPs.

8. Increasing the computational efficiency of the bootstrap estimation of the test statistics null distribution.

References

Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *JRSSB*, 57:289–300, 1995.

Y. Benjamini and D. Yekutieli. The control of the false discovery rate in multiple hypothesis testing under dependency. *The Annals of Statistics*, 29(4):1165–1188, 2001.

J. M. Chambers. *Programming with Data: A Guide to the S Language*. Springer-Verlag, New York, 1998.

S. Chiaretti, X. Li, R. Gentleman, A. Vitale, M. Vignetti, F. Mandelli, J. Ritz, and R. Foa. Gene expression profile of adult t-cell acute lymphocytic leukemia identifies distinct subsets of patients with different response to therapy and survival. *Blood*, 103(7):2771–2778, 2004.

S. Dudoit and M. J. van der Laan. *Multiple Testing Procedures and Applications to Genomics*. Springer, 2004. (In preparation).

S. Dudoit, M. J. van der Laan, and K. S. Pollard. Multiple testing. Part I. Single-step procedures for control of general Type I error rates. *Statistical Applications in Genetics and Molecular Biology*, 3(1):Article 13, 2004. URL `www.bepress.com/sagmb/vol3/iss1/art13`.

Y. Hochberg. A sharper bonferroni procedure for multiple tests of significance. *Biometrika*, 75:800–802, 1988.

S. Holm. A simple sequentially rejective multiple test procedure. *Scand. J. Statist.*, 6:65–70, 1979.

S. Keleş, M. J. van der Laan, S. Dudoit, and S. E. Cawley. Multiple testing methods for ChIP-Chip high density oligonucleotide array data. Technical Report 147, Division of Biostatistics, University of California, Berkeley, 2004. URL `www.bepress.com/ucbbiostat/paper147`.

K. S. Pollard and M. J. van der Laan. Choice of a null distribution in resampling-based multiple testing. *Journal of Statistical Planning and Inference*, 125(1–2):85–100, 2004.

M. J. van der Laan, S. Dudoit, and K. S. Pollard. Augmentation procedures for control of the generalized family-wise error rate and tail probabilities for the proportion of false positives. *Statistical Applications in Genetics and Molecular Biology*, 3(1):Article 15, 2004a. URL `www.bepress.com/sagmb/vol3/iss1/art15`.

M. J. van der Laan, S. Dudoit, and K. S. Pollard. Multiple testing. Part II. Step-down procedures for control of the family-wise error rate. *Statistical Applications in Genetics and Molecular Biology*, 3(1):Article 14, 2004b. URL `www.bepress.com/sagmb/vol3/iss1/art14`.

Z. Šidák. Rectangular confidence regions for the means of multivariate normal distributions. *Journal of the American Statistical Association*, 62: 626–633, 1967.

P. H. Westfall and S. S. Young. *Resampling-based multiple testing: Examples and methods for p-value adjustment*. John Wiley & Sons, 1993.